AMENDMENTS TO THE CLAIMS

Kindly replace the claims as follows.

1. (Currently amended) A computer, comprising:

[[an]] instruction pipeline circuitry configured to execute instructions of the computer;

profile hardware circuitry configured to detect and record, without compiler assistance for execution profiling, profile information describing a sequence of events occurring in the instruction pipeline, the sequence including every event occurring during a profiled execution interval that matches time-independent selection criteria of events to be profiled, the recording continuing until a predetermined stop condition is reached, the profile circuitry further configured to detect the occurrence of a predetermined condition, after a non-profiled interval of execution, and to thereon commence of the profiled execution interval.

2. (Original) The computer of claim 1:

the profile circuitry being configured to record at least one physical memory reference noting the source and destination of a control flow event in which control flow of the program execution diverges from sequential execution.

3. (Currently amended) The computer of claim 1, wherein:

the instruction pipeline circuitry is configured to execute instructions of an instruction set in which an interpretation of an instruction depends on a processor mode not expressed in a binary representation of the instruction; and

the profile circuitry is configured to record profile information describing at least all events occurring during the profiled execution interval of the two classes:

a divergence of execution from sequential execution, and

a processor mode change that is not inferable from the opcode of the instruction that induces the processor mode change, the instruction opcode taken together with a processor mode before the mode change instruction.

4. (Currently amended) The computer of claim 3, wherein:

the profile circuitry is configured to detect the occurrence of profileable events occurring in the instruction pipeline circuitry, and to direct the instruction pipeline circuitry to record profile information describing the profileable events essentially concurrently with the occurrence of the profileable events, the detecting and recording occurring under hardware control without software intervention.

5. (Currently amended) The computer of claim 3, wherein:

the profile circuitry is cooperatively interconnected with the instruction pipeline circuitry to record the profile information into general registers of the computer under hardware control, without software intervention, and without first storing the profile information into main memory.

6. (Original) The computer of claim 3, wherein the profile circuitry is configured to record profile information efficiently tailored to identify all bytes of object code executed during the profiled interval, without reference to the binary code of the program.

7. (Original) The computer of claim 1, wherein:

the profile circuitry is configured to record profile information identifying each distinct physical page of instruction text executed during the profiled execution interval.

8. (Currently amended) The computer of claim 1, wherein:

the instruction pipeline circuitry is configured to execute instructions of an instruction set in which an interpretation of an instruction depends on a processor mode not expressed in a binary representation of the instruction; and

the profile circuitry is configured to record profile information describing at least all events occurring during the profiled execution interval of the two classes:

a divergence of execution from sequential execution;

a processor mode change that is not inferable from the opcode of the instruction that induces the processor mode change, the instruction opcode taken together with a processor mode before the mode change instruction; and

the profile circuitry is configured to record profile information identifying each distinct physical page of instruction text executed during the profiled execution interval.

9. (Original) The computer of claim 8, wherein the triggering event is expiration of a timer.

10. (Currently amended) The computer of claim 1, wherein the program is executed on a computer having:

[[an]] instruction pipeline circuitry configured to execute instructions from a memory of the computer; and

profile circuitry configured to detect the occurrence of profileable events occurring in the instruction pipeline circuitry, and to direct the instruction pipeline circuitry to record profile information describing the profileable events essentially concurrently with the occurrence of the profileable events, the detecting and recording occurring under hardware control without software intervention.

11. (Original) The computer of claim 10, wherein:

the profile information is recorded into general registers of the computer under hardware control, without software intervention, and without first storing the profile information into main memory.

12. (Currently amended) The computer of claim 1:

wherein the instruction pipeline circuitry, profile circuitry, and trigger are components of a first CPU of a multiprocessor;

and further comprising a second CPU of the multiprocessor, configured to analyze the recorded profile information while the execution and profiling of the program continues on the

first CPU, and to at least partially control the operation of the first CPU based at least in part on the analysis of the collected profile data.

13. (Original) The computer of claim 12, wherein the controlling is effected by altering the text of the program executed by the first CPU.

14. (Original) The computer of claim 1, wherein the recorded physical memory references include addresses of binary instructions referenced by an instruction pointer, and at least one of the recorded instruction references records the event of a sequential execution flow across a page boundary in the address space.

15. (Amended 11/1/2000) The computer of claim 14, wherein at least one of the recorded instruction references records the event of a page boundary of the address space occurring within a single instruction.

16. (Amended 11/1/2000) The computer of claim 15, wherein at least one of the recorded instruction references records the event of a page boundary between two instructions that are sequentially adjacent in the logical address space.

17. (Original) The computer of claim 1, wherein:

the recorded profile information indicates ranges of instruction binary text executed by the computer during a profiled interval of the execution, the ranges of executed text being recorded as low and high boundaries of the respective ranges.

18. (Original) The computer of claim 17, wherein the profile circuitry is configured to record, as the high boundaries of the respective ranges, the address of the last byte of each respective range.

19. (Original) The computer of claim 1, wherein:

the captured profile information comprises subunits of two kinds, a first subunit kind describing an instruction interpretation mode at an instruction boundary, and a second subunit kind describing a transition between processor modes.

20. (Original) The computer of claim 1:

wherein the criteria for profileable events divide the profileable events into initiating events and non-initiating events;

the profile circuitry being further designed:

after the triggering event is detected, to ignore non-initiating events; and

when an initiating event is detected, to commence recording the profile entries in the memory, describing every initiating and non-initiating event matching the profileable criteria during an interval following the triggering event.

21. (Original) The computer of claim 1, the profile circuitry being further designed to record a timestamp describing a time of the recorded events.

22. (Original) The computer of claim 1, the profile circuitry being further designed to record an event code describing the class of each profileable event recorded.

23. (Original) The computer of claim 22, wherein a number of bits used to record the event code is less than $\log_2$ of the number of distinguished event classes

24. (Currently amended) The computer of claim 1:

wherein the instruction pipeline circuitry is configured to execute instructions of two instruction sets, a native instruction set providing access to substantially all of the resources of the computer, and a non-native instruction set providing access to a subset of the resources of the computer.

25. (Currently amended) The computer of claim 24, wherein the instruction pipeline circuitry and profile circuitry are further configured to effect recording of profile information describing an interval of the execution of an operating system coded in the non-native instruction set.

26. (Original) The computer of claim 1, wherein the predetermined triggering condition is the expiration of a timer.

27. (Original) The computer of claim 1, wherein the recorded profile information is efficiently tailored to identify all bytes of object code executed during the profiled interval, without reference to the binary code of the program.

28. (Currently amended) A method, comprising the step of:

during execution of instruction of instructions in [[a]] hardware pipeline circuitry of a computer, operating profile circuitry designed to detect and record, without compiler assistance for execution profiling, profile information describing a sequence of events occurring in the instruction pipeline circuitry, the sequence including every event occurring during a profiled execution interval that matches time-independent selection criteria of events to be profiled, the recording continuing until a predetermined stop condition is reached, the profile circuitry further configured to detect the occurrence of a predetermined condition, after a non-profiled interval of execution, and to thereon commence of the profiled execution interval.

29. (Previously presented) The method of claim 28, wherein:

the profile circuitry is configured to record at least one physical memory reference noting the source and destination of a control flow event in which control flow of the program execution diverges from sequential execution.

30. (Currently amended) The method of claim 28, wherein:

the instruction pipeline circuitry is configured to execute instructions of an instruction set in which an interpretation of an instruction depends on a processor mode not expressed in a binary representation of the instruction; and

the profile circuitry is configured to record profile information describing at least all events occurring during the profiled execution interval of the two classes:

a divergence of execution from sequential execution, and

a processor mode change that is not inferable from the opcode of the instruction that induces the processor mode change, the instruction opcode taken together with a processor mode before the mode change instruction.

31. (Currently amended) The computer of claim 3, wherein:

the profile circuitry is configured to detect the occurrence of profileable events occurring in the instruction pipeline circuitry, and to direct the instruction pipeline circuitry to record profile information describing the profileable events essentially concurrently with the occurrence of the profileable events, the detecting and recording occurring under hardware control without software intervention.

32. (Previously presented) The method of claim 28, wherein:

the profile circuitry is configured to record profile information identifying each distinct physical page of instruction text executed during the profiled execution interval.

33. (Currently amended) The method of claim 28, wherein:

the instruction pipeline circuitry is configured to execute instructions of an instruction set in which an interpretation of an instruction depends on a processor mode not expressed in a binary representation of the instruction; and

the profile circuitry is configured to record profile information describing at least all events occurring during the profiled execution interval of the two classes:

a divergence of execution from sequential execution;

a processor mode change that is not inferable from the opcode of the instruction that induces the processor mode change, the instruction opcode taken together with a processor mode before the mode change instruction; and

the profile circuitry is configured to record profile information identifying each distinct physical page of instruction text executed during the profiled execution interval.

34. (Currently amended) The method of claim 28, wherein the program is executed on a computer having:

[[an]] instruction pipeline circuitry configured to execute instructions from a memory of the computer; and

profile circuitry configured to detect the occurrence of profileable events occurring in the instruction pipeline circuitry, and to direct the instruction pipeline circuitry to record profile information describing the profileable events essentially concurrently with the occurrence of the profileable events, the detecting and recording occurring under hardware control without software intervention.

35. (Previously presented) The method of claim 28, wherein the recorded physical memory references include addresses of binary instructions referenced by an instruction pointer, and at least one of the recorded instruction references records the event of a sequential execution flow across a page boundary in the address space.

36. (Previously presented) The method of claim 28, wherein:

the recorded profile information indicates ranges of instruction binary text executed by the computer during a profiled interval of the execution, the ranges of executed text being recorded as low and high boundaries of the respective ranges.

37. (Currently amended) The method of claim 28:

wherein the instruction pipeline circuitry is configured to execute instructions of two instruction sets, a native instruction set providing access to substantially all of the resources of

the computer, and a non-native instruction set providing access to a subset of the resources of the computer.

38. (Previously presented) The method of claim 28, wherein the predetermined triggering condition is the expiration of a timer.